

Amendments to the Specification

The following paragraphs are sought to be amended as shown.

Please amend the paragraph spanning lines 5-18 on page 8:

The transaction management server 640b has mapping rules for each site, and serves to transform the messages from the content-site vocabulary to that of the merchant site 20. The transaction management server 640b accepts user requests, translates them, and issues the appropriate transaction request at the merchant site ~~[[20f]]~~ 20. Translation is done with the following steps: (i) the content site request is fetched according to the content site identification and page identification, (ii) the content site request is translated into the information utility 250 internal representation, (iii) the user is allowed to edit the request, (iv) the internal representation is translated into the merchant transaction form; and (v) the transaction is submitted to the merchant site 20. The merchant used depends on the user profile. The content site request is a set of objects representing categories or products. The internal representation is the same as the content site request representation. The merchant transaction form consists of a set of tuples. Each tuple contains the user identification and the SKU of the merchant product. (The user identification may need to be mapped to the merchant user identification.)

Please amend the paragraph spanning line 30 of page 9 to line 13 of page10:

A user can either access a content site 10 or merchant site 20. For example, a new user may visit the merchant site ~~[[20f]]~~ 20. When rendering the page for the user, the image server (for example, the merchant image handler of Image I 640c) determines that this user is new and displays a user registration box (not shown). A cookie is dropped on the user browser 610 at this time. The user selects an option on the user registration box and is routed by the click server (for example, the merchant click handler of Click 1, 640e) to the user registration system 640g. The user registration system 640g registers the user with the information utility 250. At a later time, the user may visit a content site 10. The user selects the image and is routed by the content click handler ~~[[941]]~~ 640f to the transaction management server 640b. The transaction management server 640b

translates the request associated with the content page 10 into the appropriate transaction for the associated merchant site 20. The transaction is issued to the merchant site 20 and the webpage is redisplayed. Upon redisplay, the content image handler of 640c or d detects that the user has issued a transaction and displays the appropriate image. At a later time, the user visits the merchant site 20 and sees the result of the transaction.

Please amend the paragraph spanning lines 4-13 on page 11:

Referring first to the user image delivery phase, and referring to FIGS. 3a and 3b, a user may access either a merchant site 20 or content site 10 via a user browser 610 such as Microsoft Internet Explorer or Netscape Navigator. However, the invention is not limited to use of a browser. The invention applies equally to the use of wireless devices. If the user accesses the content site 10, the user browser 610 first requests an HTML (Hypertext Markup Language) page from the content site 10 by generating an HTTP (Hypertext Transfer Protocol) message (path [[243]] 241). (910) The content site 10 responds with an HTML page having an embedded image reference to information utility 250, which identifies the recipe (path [[241]] 243). (920) If wireless devices are employed, a wireless access protocol may be used instead of HTTP.

Please amend the paragraph spanning lines 14-22 on pages 11:

Next, the user clicks on the embedded image and sends a request to the information utility 250. For example, the user browser 610 may request the recipe image from the information utility 250 via path 257 and may send the information utility 250 an information utility cookie if the user is a registered user. [(930).] For example, the user browser 610 requests the recipe image from the image server (for example content image handler of Image 1, 640c, FIG. 6) component of the information utility 250 via the HTML fragment:

 where 4 is an example pre-assigned content identifier and 5 is a pre-assigned recipe identifier.

Please amend the paragraph spanning lines 22-25 on page 12 :

Next, the user may issue a transaction by sending a request to information utility 250 with his or her information utility cookie. The user may click through on the image (e.g., shopping icon 430), sending a request to a click server (for example, the content click handler of Click 1, 640e, FIG. 6) ~~component~~ via the HTML fragment:

Please amend the paragraph spanning line 26 of page 14 to line 5 of page 15:

Next, the information utility 250 receives a request from the user browser 610 for an image. The user browser 610 contacts the information utility 250 with merchant supplied parameters including the mui. (730) The information utility 250 then determines if an information utility 250 cookie exists. If it does, then information utility 250 drops a cookie, and logs the new user in its database server 640a (FIG. 6). (740) No attached cookie implies that this request comes from a new user. In that event, the handler generates a new information utility 250 user object, assigns an identifier, say "3," and saves a tuple in P (734). Thus, the triple (informationutility-id=3, merchant-id=1, merchant-user-id=2) is saved. A single-pixel image is returned as the response to the request. ~~Then the information utility 250 drops a cookie and logs the user into its database (740) a cookie (informationutility-id=3) is dropped at the user browser 610.~~ Then the information utility 250 drops a cookie (informationutility-id=3) at the user browser 610 and logs the user into its database (740).

Please amend the paragraph spanning line 26 of page 16 to line 3 of page 17:

In another example, a user signs up with merchant "1" using browser 610a and inserting tuple (3,1,2) into P. The same user may sign up with merchant "2" using browser 610i and inserting tuple (4,2,3) in P. (The above protocol is simply executed twice here.) These two users will be treated as two independent users. However, subsequently, when the user signs up with merchant "2" using browser 610a or merchant "1" using browser 610i, the two users are collapsed into a single user. This collapsing operation is accomplished by replacing the existing cookie with the matched information utility [[951]] identifier and all tuples containing the existing cookie identifier are replaced with the matched information utility 250 identifier in P (Rule 2).